T.C.
MİLLÎ EĞİTİM BAKANLIĞI

YENİLENEBİLİR ENERJİ TEKNOLOJİLERİ

PARALEL PORT KONTROLÜ

Ankara, 2015

- Bu modül, meslekî ve teknik eğitim okul/kurumlarında uygulanan Çerçeve Öğretim Programlarında yer alan yeterlikleri kazandırmaya yönelik olarak öğrencilere rehberlik etmek amacıyla hazırlanmış bireysel öğrenme materyalidir.
- Millî Eğitim Bakanlığınca ücretsiz olarak verilmiştir.
- PARA İLE SATILMAZ.

İÇİNDEKİLER

AÇIKLAMALAR ii
GİRİŞ 1
ÖĞRENME FAALİYETİ-1 3
1. PARALEL PORTUN TANIMI 3
1.1. Bios Parallel Port Tablosu 3
1.2. Paralel port Çeşitleri 5
1.2.1. SPP (Standard Parallel Port) 5
1.2.2. Bi-Directional 5
1.2.3. EPP (Enhanced Parallel Port) 5
1.2.4. ECP (Extended Capability Port) 6
1.3. Veri yazmacı 6
1.4. Durum Yazmacı 6
1.5. Denetim Yazmacı 7
UYGULAMA FAALİYETİ 8
ÖLÇME VE DEĞERLENDİRME 13
ÖĞRENME FAALİYETİ–2 15
2. PARALEL PORTTAN ÇIKIŞ ALMAK 15
2.1. Paralel Portun Veri Yazmacından Çıkış Almak 16
2.2. Paralel Portun Denetim Yazmacından Çıkış Almak 18
UYGULAMA FAALİYETİ 21
ÖLÇME VE DEĞERLENDİRME 28
ÖĞRENME FAALİYETİ–3 30
3. PARALEL PORTTAN GİRİŞ ALMAK 30
3.1. Paralel Port Durum Yazmacından Giriş Almak 30
3.2. Paralel Port Veri Yazmacından Giriş Almak 32
UYGULAMA FAALİYETİ 36
ÖLÇME VE DEĞERLENDİRME 39
MODÜL DEĞERLENDİRME 41
CEVAP ANAHTARLARI 44
KAYNAKÇA 45

AÇIKLAMALAR

ALAN	**Yenilenebilir Enerji Teknolojisi**
DAL/MESLEK	**Bilgisayralı Kontrol**
MODÜLÜN ADI	**Paralel Port Kontrolü**
MODÜLÜN TANIMI	Paralel portun yapısı ve visual basic programlama dilinde paralel port üzerinden veri alışverişi işlemlerini yapabilme yeterliliğinin kazandırıldığı modüldür.
SÜRE	40/32
ÖN KOŞUL	
YETERLİK	Paralel port kontrolü yapmak.
MODÜLÜN AMACI	**Genel Amaç:** Paralel port ile giriş ve çıkış işlemlerini doğru olarak yapabileceksiniz. **Amaçlar** 1. Paralel port kontrol devresini devre şemasına ve baskı devre tekniklerine uygun olarak yapabileceksiniz. 2. Paralel port donanımı yoluyla sayısal çıkış değerlerini kontrol edebileceksiniz. 3. Paralel port donanımı yoluyla sayısal giriş değerlerini kontrol edebileceksiniz.
EĞİTİM ÖĞRETİM ORTAMLARI VE DONANIMLARI	**Ortam:** Bilgisayar Laboratuarı, Elektrik-Elektronik Laboratuarı **Donanım:** Visual Basic 6.0 veya Microsoft Visual Basic 2010 Express çalıştırabilen bilgisayar, baskı devre araç gereçleri, lehimleme araç gereçleri, multimetre **Yazılım:** Visual Basic 6.0 veya Microsoft Visual Basic 2010 Express programlama dili
ÖLÇME VE DEĞERLENDİRME	Her faaliyetin sonunda ölçme soruları ile öğrenme düzeyiniz ölçeceksiniz. Araştırmalarla grup çalışmaları ve bireysel çalışmalarla öğretmen rehberliğinde ölçme ve değerlendirmeyi gerçekleştirebileceksiniz.

GİRİŞ

Sevgili Öğrenci,

Bir kumanda işlemini bilgisayarla yapmaya karar veren bir teknisyen için paralel port iyi bir seçenek olabilir. Paralel porta hükmedebilmek için ise onun yapısının iyi bilinmesi gerekmektedir. Esasında yapısı oldukça basit olan paralel port iyi bilindiğinde herhangi bir programla dili kullanılarak etkin uygulamalar geliştirilebilir. Elbette tercih edilecek programlama dilinin paralel port kontrol ve erişim komutlarının iyi bilinmesi de gerekmektedir.

Bu modülün birinci bölümünde yazıcı bağlantı noktası olarak geliştirilen paralel port etraflıca incelenerek bilinmesi gereken temel hususlara değinilmiştir. İkinci bölümde ise Visual Basic programlama dili kullanılarak paralel port üzerinden bilgilerin nasıl dış ortama gönderileceği konusu işlenmiştir. Son bölüm ise paralel port kullanılarak dış ortamdan gönderilen işaretlerin alınması konusuna ayrılmıştır.

Her bölümün sonunda uygulama faaliyetleri verilmiştir. Bu uygulama faaliyetlerinde program kodları sizleri kendi algoritmalarınızı serbestçe düşünmeye ve oluşturmaya sevk etmek için özellikle verilmemiştir.
Bu üç bölümde verilen örnek uygulamaların eksiksiz yapıldığı takdirde paralel portla bilgi alışverişi yönteminin zihinlerinizde aydınlanacağına inanıyoruz.

ÖĞRENME FAALİYETİ-1

AMAÇ

Paralel port kontrol devresini devre şemasına va baskı devre tekniklerine uygun olarak yapabileceksiniz.

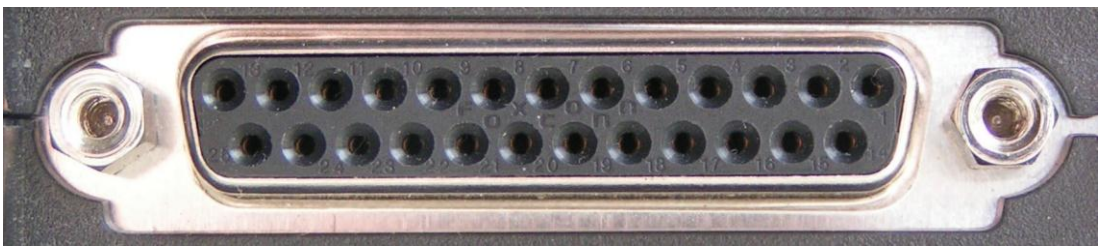
ARAŞTIRMA

- Endüstride, okulunuzdaki atölye ve laboratuarlarda paralel port kullanarak haberleşme yapan cihaz ya da sistemleri araştırınız.

1. PARALEL PORTUN TANIMI

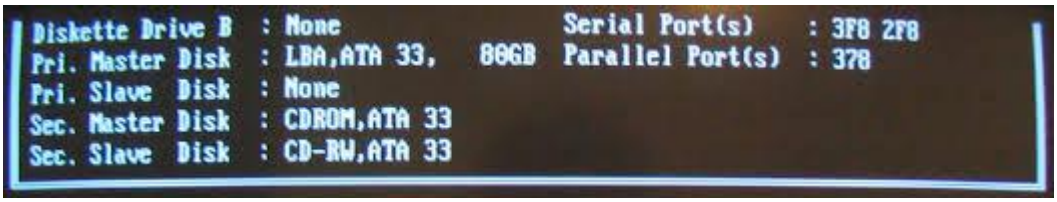
1.1. Bios Parallel Port Tablosu

Bilgisayar denilince insanlar genellikle hızla hesap yapan, bilgileri tutan ve gösteren, İnternet'e girilebilen bir büro cihazı düşünür. Bu düşünce doğru olmakla birlikte bilgisayarların işlevlerinin bir kısmını oluşturur. Bilgisayarlar aynı zamanda farklı ekipmanlar ve makinelerle de irtibat kurabilen, haberleşebilen ve onları kontrol edebilen cihazlardır.



Resim 1.1: Paralel port konnektörü (dişi)

Bilgisayarın I/O haritasında yer alan paralel port, seri port benzeri port adresleri ile takılı olan kartların adresleri BIOS tarafından okunarak bir tablo hâline getirilir. Bu her bilgisayar açılışında karşımıza çıkmaktadır.



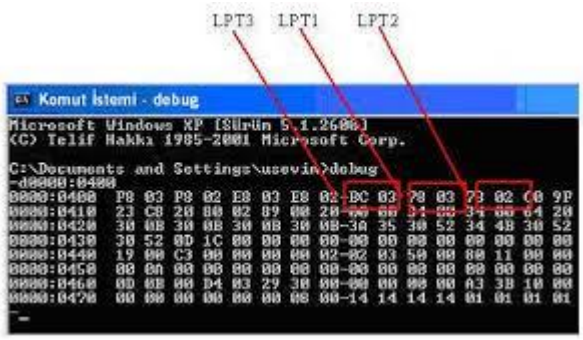
Resim 1.2: Bilgisayar açılış ekranı

BIOS tablosunda tutulan adreslerin yerleri standarttır. Dolayısıyla bu tablonun 0x400 :0008 adresinden itibaren LPT1, LPT2 ve LPT3 portlarından var olanlarının adresleri tutulur. Windows komut istemini çalıştırarak "Debug" işlemiyle görebiliriz.

Bunun için komut istemine şu komutları uygulayalım:

```
C:\debug
- d0000:0400
```

Bu işlemler sonunda aşağıdaki şekilde görüldüğü gibi karşımıza onaltılık sayıtabanında adresleme tablosu gelir. Farklı bir bilgisayarda aynı tabloyu görmeye çalıştığımızda adres sıralarının boş ya da sırasının değişik olduğunu görebiliriz. Diğer bir noktada bilgisayarımızda tek bir tane paralel port olmasına karşın bu tabloda üç tane adresin var olduğunu görebiliriz. Bu durumlar BIOS versiyonlarından kaynaklanır.



Resim 1.3: Debug işlemi ve adresler

Paralel port konnektöründe bulunan pinlerin görevleri aşağıdaki şekildedir:

Pin Numarası	Görevi	Açıklama
1.pin	Metronom(strobe) hattı.	
2-9. pinler	Veri hattı	
10. pin	Acknowledge (hazır) hattı	Yazıcının hazır olduğu ve bilgilerin gönderebileceğini belirtir.
11. pin	Busy (Meşgul) hattı	
12. pin	No paper (Kâğıt bitti) hattı	
13. pin	Selected (seçme) hattı	
14. pin	AutoLineFeed (Satırbaşı) hattı	
15. pin	Error (Hata) hattı	
16. pin	İnitialize (Açma/Kapama) hattı	
17. pin	Select (Hazır / Değil) hattı	
18-25. pinler	Ground (Toprak) hattı	

1.2. Paralel port Çeşitleri

İlk tasarlanan paralel portlardan sonra zaman içerisinin de farklı ve daha gelişmiş port yapıları ortaya çıkmıştır. Her yeni çıkan yapı kendinden önceki yöntemleri Destekle-mektedir. Bu farklı çalışma yapıları **mod** olarak isimlendirilmiştir.

1.2.1. SPP (Standard Parallel Port)

Yazıcı bağlantısını gerçekleştirmek amacıyla ilk olarak geliştirilmiştir. SPP modunda paralel portun data yazmacı yalnızca çıkış olarak kullanılmaktadır.

1.2.2. Bi-Directional

Yazıcı ile çift yönlü haberleşmeye imkân verecek şekilde tasarlanan portlardır. Yani data hattı hem çıkış hem de giriş olarak kullanılmaktadır.

1.2.3. EPP (Enhanced Parallel Port)

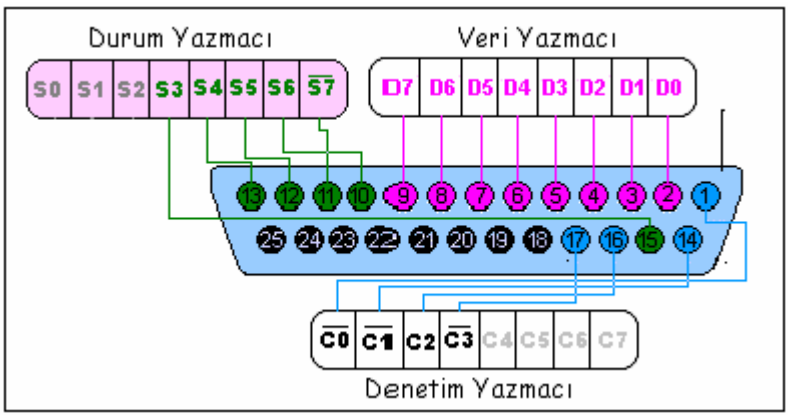
Geliştirilmiş paralel port anlamına gelen EPP modunda bilgisayar ile yazıcı önceki modlarda programcılar tarafından yazılım yoluyla yaptırılan el sıkışma (handshaking) adı verilen işlemi donanım olarak gerçekleştirebilmektedir. El sıkışma işleminde kablo bağlantısı kurulduğunda her iki taraftaki cihazlar kendi durumları ile ilgili bilgileri birbirlerine göndererek haberleşmenin doğru ve güvenilir bir şekilde başlamasını sağlarlar ayrıca EPP modu daha hızlı haberleşmenin de yolunu açmıştır.

1.2.4. ECP (Extended Capability Port)

Geliştirilmiş kapasiteli port anlamına gelen ECP modunda EPP modunun getirmiş olduğu yeniliklerin yanı sıra tek gövde içerisinde birden fazla barındıran büro cihazlarıyla haberleşme imkânı sunar.

1.3. Veri yazmacı

Paralel port üç adet yazmaç (register) tarafından kontrol edilir. Bu yazmaçla-rın her birinin kendine ait bir adresi vardır. Bu adresler vasıtasıyla programcı yaz-maçlara ve dolayısıyla paralel porta ulaşabilir.



Resim 1.4: Paralel port yazmaçları

Yükseklik ve genişlik değerleri eşit olursa dikdörtgen, kareye dönüşecektir. Komutla dkdörtgen çizdirilebilir. Aşağıdaki örnek 5 ve 6 incelenirse komutların nasıl kullanıldığı anlaşılacaktır.

Veri yazmacı konnektör üzerinden gönderilecek veya alınacak 8 adet veri bitini (D0,D7) tutar. Yazmaca yeni bir bilgi yazılıncaya kadar bu tutma işlemini yapar. Bu yazma bilgisayardan yazıcıya gönderilen belgelerin aktarılması işinde kullanılır. Veri yazmacı bitleri konnektörün 2 ile 9. pinleri arasında fiziksel karşılığını bulur. Herhangi bir bilgisayarlı kontrol sisteminde genellikle dışarıya veri gönderme işlemlerinde tercih edilir.

1.4. Durum Yazmacı

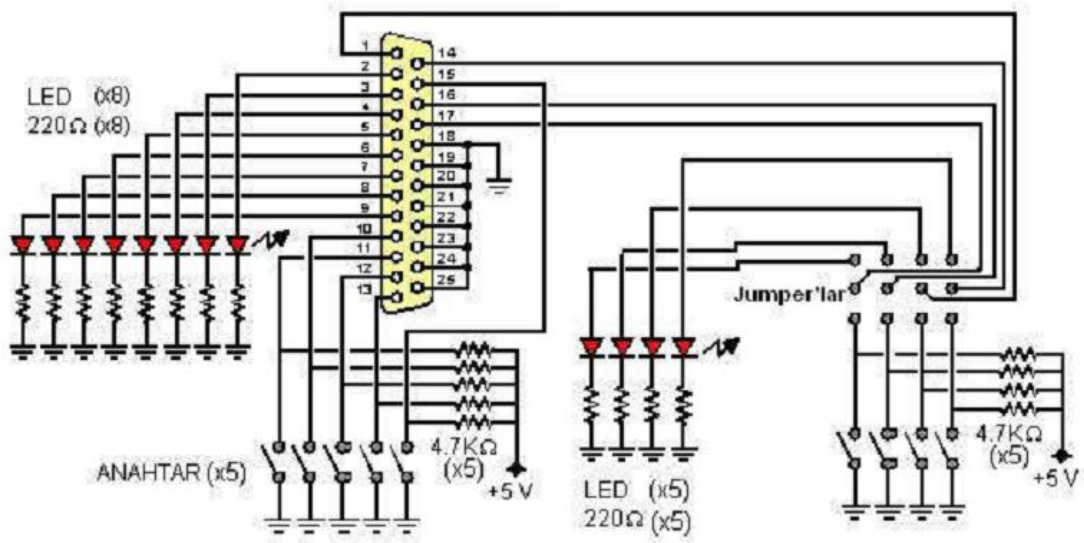
Esas görevi haberleşme esnasında yazıcının durumuna ilişkin gelen sinyalleri tutmak olan durum portu aynı zamanda harici ortamdan giriş sinyali almak için de kullanılabilir. 10,11,12,13,14 ve 15 no'lu pinlere bağlı 5 biti kullanılan bu yazmacın diğer 3 biti kullanılmamaktadır. 11 no'lu pine bağlı S7 biti terlenmiş durumdadır. Kullanılmayan 3 bit ileride geliştirilebilecek yöntem ve tekniklerde kullanılmak üzere ayrılmıştır.

1.5. Denetim Yazmacı

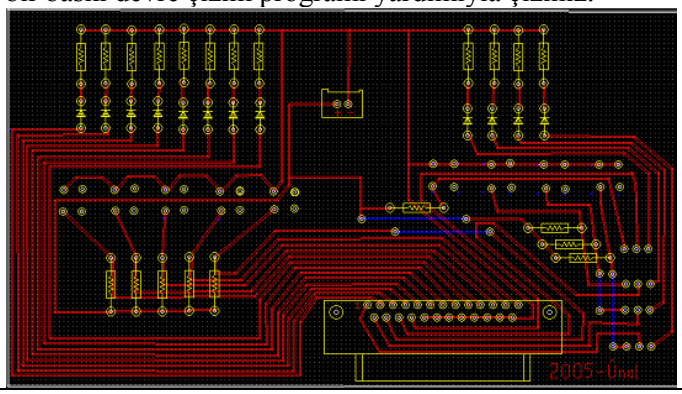
Bilgisayarın yazıcıyı kontrol edebilmesi için gerekli sinyalleri tutan bu yaz-maç hem giriş hem de çıkış olarak harici ortamdan sinyal alışverişinde kullanılır. Port konnektörü üzerindeki 4 adet pin bu yazmacın 4 bitine bağlanmıştır. Bu bitler-den C0, C1 ve C3 bitleri terslenmiştir. Diğer 4 bit ise durum portunda olduğu gibi ileride geliştirecek uygulamalarda kullanılmak üzere ayrılmıştır.

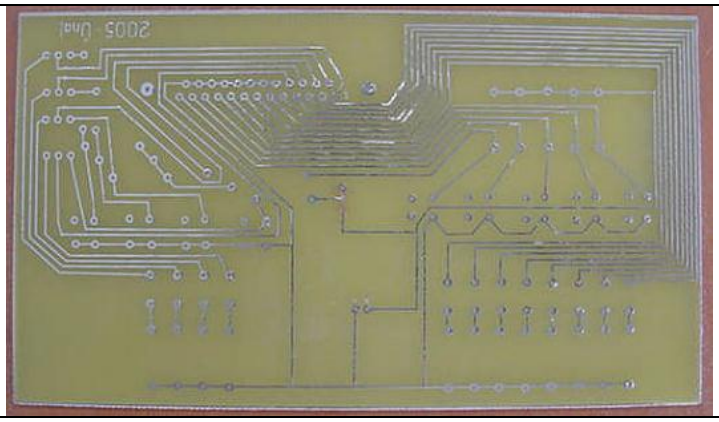
UYGULAMA FAALİYETİ

Aşağıdaki işlem basamaklarına göre uygulama faaliyetini yapınız.

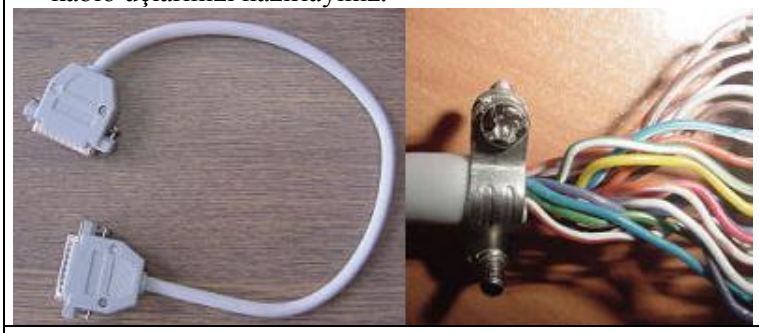


Resim 1.5: Deney kartı devresi

İşlem Basamakları	Öneriler
➢ Yukarıda verilen devrenin baskı devre şemasını, her hangi bir baskı devre çizim programı yardımıyla çiziniz. 	
➢ Çizdiğiniz baskı devre şemasını pozitif 20 ya da ipek baskı gibi seçeceğimiz bir yöntem ile bakır plaket üzerine aktarınız.	

	
➢ Hazır olan bakır plaketimizi çözelti içerisine atarak baskı devre kartımızı elde ediniz. 	
➢ Elemanların montajı için gerekli delikleri matkap yardımıyla açınız.	0,8 mm'lik matkap ucu kullanınız.

➢ Lehim kullanarak tüm elemanları kart üzerine monte ediniz. 	
➢ Deney kartı üzerine elemanların montajının tamamlan masının ardından pleksiglas malzemeden altlık keserek katın tabanını ve ayaklarını monte ederek işlemi tamamlayınız.	

➢ Bağlantı kablosu için yeterli uzunlukta kablo kesiniz ve kablo uçlarınızı hazırlayınız. 	
➢ Kablonun her iki tarafına 25 iğneli konnektörlerin montajını yaparak işlemi sonlandırınız. 	Kabloların montajında telleri bire bir lehimleyiniz. Örneğin, A Konnektörünün 2. pinindeki teli B konnektörünün 2. piniyle lehimleyiniz.

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
1. Baskı devre şemasını çizebildiniz mi?		
2. Baskı devreyi bakır plaket üzerine aktarabildiniz mi?		
3. Deney kartını üretebildiniz mi?		
4. Deney kartını kablosunu deney kartına bağlayabildiniz mi?		
5. Deney kartı kablosunu parelel porttaki doğru pinlere bağlayabildiniz mi?		

DEĞERLENDİRME

Değerlendirme sonunda "Hayır" şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız "Evet" ise "Ölçme ve Değerlendirme" ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları dikkatlice okuyunuz ve doğru seçeneği işaretleyiniz.

1. Aşağıdakilerden hangisi bir port değildir?
   A) IrDA
   B) Seri
   C) Status
   D) PS/2

2. Paralel port yapısından kaç tane yazmaç bulunur?
   A) 1
   B) 2
   C) 3
   D) 4

3. Geliştirilmiş kapasiteli port tanımı aşağıdakilerden hangisine uyar?
   A) PPP
   B) SPP
   C) EPP
   D) ECP

4. Durum yazmacının görevi nedir?
   A) Gönderilecek bilgileri denetler.
   B) Yazıcının durumuna ilişkin gelen sinyalleri tutar.
   C) İletişim esnasında bilgisayarın durumuna ilişkin verileri tutar.
   D) Bilgisayarın yazıcıyı kontrol edebilmesi için gerekli sinyalleri tutar.

5. Paralel port konnektöründeki 15 no'lu pinin görevi nedir?
   A) Error (Hata)
   B) Busy (Meşgul)
   C) Ground (Toprak)
   D) Select (Seçme)

6. Aşağıdakilerden hangisi paralel port adresi değildir?
   A) 378H
   B) 278H
   C) 3F8H
   D) 03BCH

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ–2

AMAÇ

Paralel port donanımı yoluyla sayısal çıkış değerlerini kontrol edebileceksiniz.

ARAŞTIRMA

- Farklı programlama dillerinde paralel port üzerinden dış ortama veri gönderimi nasıl yapılmaktadır?

2. PARALEL PORTTAN ÇIKIŞ ALMAK

Bir bilgisayarın paralel portunu kullanarak dışarıya bilgi göndermek kolay sayılabilecek bir işlemdir. Bunu herhangi bir programlama dili kullanarak yapabili-riz. Bu bölümde Visual Basic programlama dilini gerçekleştireceğiz.

Visual Basic Port Denetimi: Dil dosyaları, programların dinamik olarak bağlantı kurabileceği fonksiyonların bir kütüphanesidir. Visual Basic programlama diline ait "inpout32.dil" kütüphanesi de içinde paralel porta erişim ve denetim için gerekli fonksiyonları içerisinde barındırır. Böylelikle programcıların fazladan gereksiz kod yazımı yapmasına gerek kalmamış olur. Inpout32.dil kütüphanesi prog-ramcının kendisi tarafından "c" ya da "Delphi" benzeri bir programla yazılabildiği gibi İnternet benzeri bir kaynaktan hazır olarak temin edilebilir.

Visual Basic kodları ile paralel porta erişmek ve denetimini yapmak için öncelikle "inpout32.dil" dosyasının bilgisayarınızın "c:\windows\system" klasörü içerisinde bulunuyor olması gerekmektedir. Eğer bilgisayarınızda bu Visual Basic eklenti dosyası mevcut değil ise gerekli dosyayı edinerek ilgili klasöre kopyalamamız gerekir.

Paralel port yazmaçlarından bir değer göndermek için "inpout32.dil" kütüp-hanesine ait "OUT( )" komutunu kullanmak yeterlidir. Bu komutun kullanım formatı aşağıdaki gibidir:

OUT Val (&adres) , Gönderilecek Veri

2.1. Paralel Portun Veri Yazmacından Çıkış Almak

Basit düzeyden karmaşığa doğru gidecek şekilde aşağıdaki örnekler yardımıyla paralel po**rttan çıkış alma işlemini öğrenelim:**

Örnek 1:	Bir metin kutusuna girdiğimiz 0-255 arası bir sayıyı paralel port üzerinden deney kartımıza gönderelim. Gönderdiğimiz verinin ikilik düzendeki karşılığını kart üzerinde bulunan veri hatlarına bağlı ledler üzerinde görelim. Bunun için aşağıda gösterilene benzer bir form tasarlayalım. Ardından kullandığımız nesnelerin özelliklerini değiştire-lim.
Ekran Görüntüsü	
Kodlar	

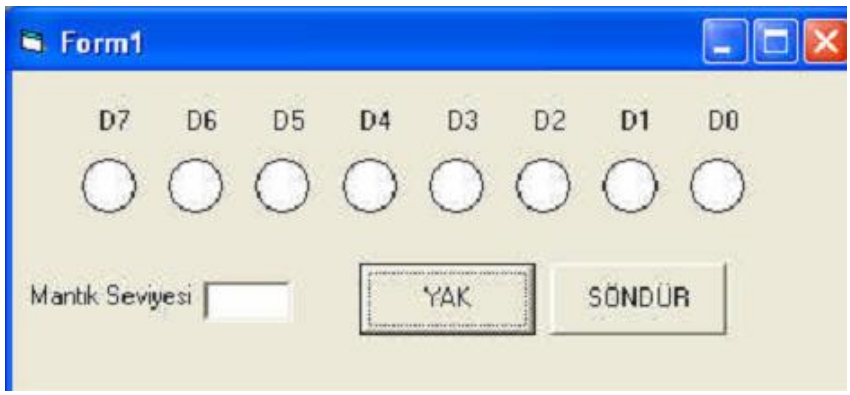
Form tasarımı bittikten sonra yeni bir modül açarak **inpout32.dll** kütüphanesini kullanabilmemiz için gerekli tanımlamaları aşağıdaki gibi yazalım. Daha sonra Module1 olarak kaydedelim.

```
Public Declare Function Inp Lib "inpout32.dll" _
Alias "Inp32" (ByVal PortAddress As Integer) As Integer
Public Declare Sub Out Lib "inpout32.dll" _
Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As Integer)
```

Program kodumuz oldukça kısa ve basittir. **If** şart yapısı ile **0-255** arası sayı girilip girilmediği denetlenmektedir. Eğer 255'ten daha büyük bir sayı girilmiş ise MsgBox fonksiyonu ile kullanıcıya bir uyarı mesajı verdirilmektedir. 255 veya daha küçük bir rakam girilmiş ise Out Val komutu ile 0378H adresindeki paralel porta metin kutusuna girilen değer gönderilmektedir. Sonuç olarak da deney kartı üzerinde bulunan 8 adet led vasıtasıyla gönderdiğimiz sayının ikili düzende karşılığını görebiliriz.

Örnek 2:	Deney kartımızın üzerinde bulunan veri hatlarına bağlı led'lerden birincisini yakan ve söndüren bir program yapalım. Bunun için aşağıda gösterilene benzer bir form tasarlayalım.

Ekran Görüntüsü



Kodlar

Aşağıdaki programı incelediğimizde "veri_portu" isimli bir alt yordamın hazırlanmış olduğunu görebiliriz:

Yak ve söndür butonlarından biri tıklandığında ilgili butona ait olan kodlar çalışır. Yak butonu tıklandığında led1 değişkeninin içeriği 1 olur.

Daha sonra veri_portu altyordamı çağrılır.

Altyordam içerisinde eğer led1 değişkeninin içeriği 1 ise text2 değişkenine 1 yazdırılır. Son olarak Out Val komutu ile 378H adresindeki paralel porta led1 içeriği olan 1 değeri (binary karşılığı 00000001'dir.) gönderilir.

Bu ise D0 ucuna bağlı olan deney kartındaki 1 no'lu led'e mantık 1 değerinin (+5V) gönderilmesi anlamına gelir ve bunun sonucunda 1 no'lu led yanar.

Bu aşamadan sonra altyordam sonlanır ve program akışı tekrar yak butonu yordamına geri döner.

Yak butonu yordamının son satırında form üzerinde D0 olarak temsil edilen led şeklinin arka planı kırmızı olur ve program sonlanır.

Benzer şekilde söndür butonu tıklandığında ise led1 değişkenine 0 bilgisi atanarak veri_portu alt yordamı çağrılır. Led1 içeriği text2 metin kutusuna yazdırıldıktan sonra led1 bilgisi paralel port üzerinden deney kartına gönderilir.

Led1 değişkeninin içeriği 0 olduğu için denet kartı üzerindeki 1 no'lu led bu defa mantık 0 (0V) bilgisi alarak söner.

```
Dim led1 As Integer ‘led1 değişkenini tüm alt yordamlarda kullanılabilecek şekilde tanım
Public Sub Veri_Portu()
Dim veri As Integer
veri=0
If led1 = 1 Then veri = 1 ‘Eğer led1 içeriği 1 iser veri değişkeninide 1 yap.
Text2.Text = veri ‘veri değişkeninin içeriğini metin kutusuna yazdır.
Out Val(&H378), Val(led1) ‘378H adresindeki paralel porta led1 içeriğini gönder.
End Sub
Private Sub sön_Click()
led1 = 0 ‘ led1 değişkeninin içeriğini 0 yap.
Call Veri_Portu ‘veri_portu alt programı çağır.
Shape2.BackColor = &HFFFFFF ‘ Form üzerindeki D0 ledini beyaz renk yapar.
End Sub
Private Sub yak_Click()
led1 = 1 ‘ led1 değişkeninin içeriğini 1 yap.
Call Veri_Portu ‘veri_portu alt programı çağır.
Shape2.BackColor = &HFF& ‘ Form üzerindeki D0 ledini kırmızı renk yapar.
End Sub
```

Program kodumuz oldukça kısa ve basittir. **If** şart yapısı ile **0-255** arası sayı girilip girilmediği denetlenmektedir. Eğer **255'**ten daha büyük bir sayı girilmiş ise **MsgBox** fonksiyonu ile kullanıcıya bir uyarı mesajı verdirilmektedir. 255 veya daha küçük bir rakam girilmiş ise **Out Val** komutu ile **0378H** adresindeki paralel porta metin kutusuna girilen değer gönderilmektedir. Sonuç olarak da deney kartı üzerinde bulunan 8 adet led vasıtasıyla gönderdiğimiz sayının ikili düzende karşılığını görebiliriz.

2.2. Paralel Portun Denetim Yazmacından Çıkış Almak

Paralel portun denetim yazmacı hem okunabilir hem de yazılabilir bir yaz-maçtır. Dört bitinden üçü terslenmiştir. Bu bacakları "mantık 1" yapmak, ilgili ba-cakta 0V olarak görülür. Denetim yazmacı normal durumda çıkış konumundadır.

Örnek 3:	Denetim yazmacına bağlı ledleri metin kutusundan gireceğimiz sayı kadar yakıp söndüren bir programın ekran görüntüsü ve koları aşağıdaki gibidir. Programı çalıştırmadan önce deney kartımızın üzerinde bulunan 4 adet köprüyü (jumper) uygun konuma getirmemiz gerekir. Bu işlemden sonra programımızı çalıştırdığımızda deney kartı üzerinde denetim yazmacına bağlı olan ledlerin yanıp söndüğünü hem kart üzerinde hem de ekran üzerinde görebiliriz.

Ekran Görüntüsü

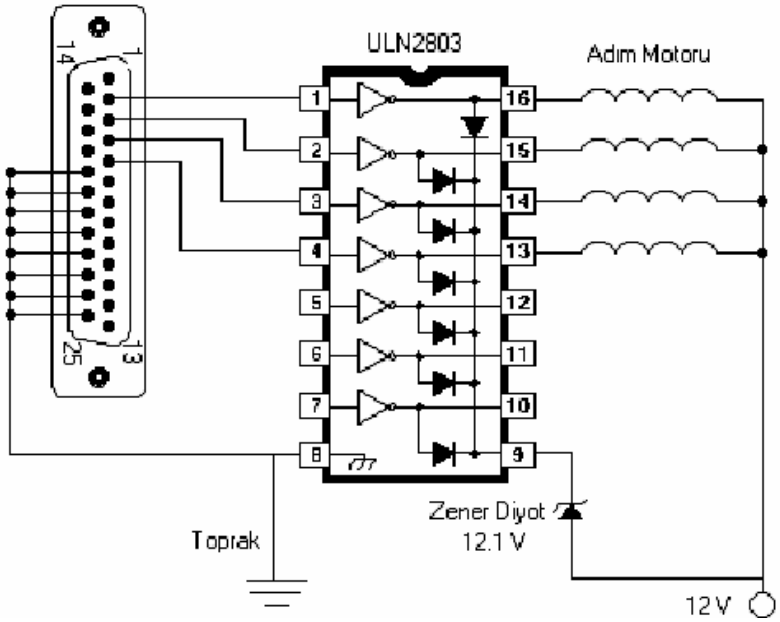
Kodlar

```
Public Sub denetim_Portu()

tüm_yak = 4 'c0,c1,c3 uçları terslendiği için 0100 bilgisi gönderilir.

tüm_sön = 11 'c0,c1,c3 uçları terslendiği için 1011 bilgisi gönderilir.

For i = 1 To Val(Text1.Text) ' for döngüsü 1'den metin kutusuna girilen değer kadardır.

Shape(0).BackColor = &HFF&

Shape(1).BackColor = &HFF&

Shape(2).BackColor = &HFF&

Shape(3).BackColor = &HFF&

DoEvents
```

```
Out Val(&H37A), tüm_yak 'denetim yazmacının adresi 37A'ya 0100 gönderir.
Call geciktir
Out Val(&H37A), tüm_sön 'denetim yazmacının adresi 37A'ya 1011 gönderir.
Shape(0).BackColor = &HFFFFFF
Shape(1).BackColor = &HFFFFFF
Shape(2).BackColor = &HFFFFFF
Shape(3).BackColor = &HFFFFFF
DoEvents
Call geciktir.
Next
End Sub
Private Sub Command1_Click()
Call denetim_Portu 'alt yordamı çağırır.
End Sub
Public Sub geciktir() 'geciktirme yordamıdır.
For j = 1 To 45000000
Next
End Sub
```

UYGULAMA FAALİYETİ

Aşağıdaki işlem basamaklarına göre uygulama faaliyetini yapınız.



Burada adım motorunu sürecek olacak uçlar veri hattının 0-3 no'lu (2-5 no'lu pinler) uçlarıdır. ULN2803 Darlington bağlantılı bir sürücü olup 500 mA akım tedarik edebilmektedir. Besleme hattının toprağı ile bilgisayarın toprak hattının birleştirildiğine dikkat ediniz.

Gerekli Bilgi

Adım Motoru: Adım motorların, endüstriyel ve elektronik uygulamalarda kullanımı oldukça fazladır. En basitinden bilgisayarımızdaki floppy disket sürücüsünde ve hard diskler de bu teknolojiye başvurulmuştur. Adım motorlar, girişlerine uygulanan lojik sinyalleri dönme hareketine çevirir. İstediğiniz yönde ve derecede döndürebileceğiniz adım motorlar, hassas hareketleri sayesinde birçok cihazda konum kontrolü amacıyla kullanılmaktradır. Dışarıdan bakınca çok komplike bir iş gibi gözükse de PC ile adım motor kontrolü oldukça kolay ve eğlencelidir**.**

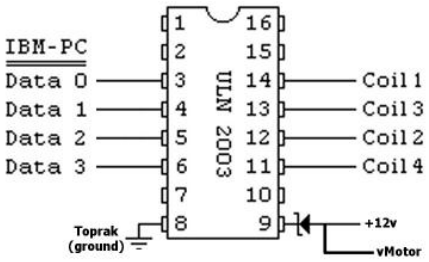
Adım motorun kablolarından bir veya iki tanesi ortaktır(vMotor). Yaptığımız işlem basit olarak bu ortak kabloya sürekli +12 Volt göndermek ve diğer uçları ise belli bir sırada toprağa göndererek bir adım hareketi elde etmek. **Bunu iki farklı şekilde yapabiliriz:** Bunlardan bir geçen ayki yazımızda kullandığımız 12 Voltluk röle devresine benzer bir devrenin 4 transistörlüsü ile bu sinyalleri göndermek. Bir diğer yolu ise içerisinde bu transistörleri hâli hazırda bulunduran bir entegre kullanmak.

Adım motor entegreli sürücü devresi malzemeleri:

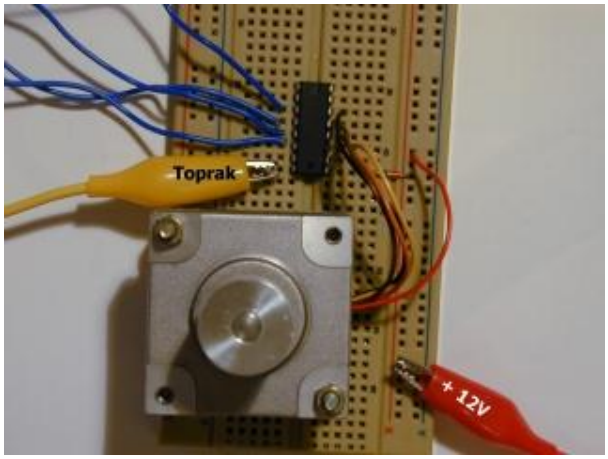
1. 1 adet ULN2003 Entegre
2. 1 adet 1N4001 Diyot
3. 1 adet 12 Volt 5 Cable Step Motor

Adım motor sürücüsü olarak ULN2003 entegresini kullanıyoruz. Sürücü devresi olarak kullanılan ULN2003 içerinde 7 adet NPN transistör ve dâhilî diod barındırdırmaktadır. Hâliyle bizi transistör bacaklarıyla uğraşmaktan kurtarmaktadır. Kullanımı ise oldukça kolaydır. Devre şemasından da anlaşılabileceği gibi 9 numaralı bacağına +12 Volt ve 8 numaralı bacağınada toprak (ground) uyguluyoruz. Daha sonra 3 ve 6 numaralı bacaklarada paralel portun DATA pinlerinden gelen +5 Voltluk değerleri uygulayacağız. Bu sayede örneğin, 3 numaralı bacağa +5 Volt (lojik voltaj) uyguladığımızda 14 numaralı bacak toprak olacaktır. Aynı şekilde sırayla 4 için 13; 5 için 12; 6 için ise 11 numaralı bacaklar toprak olacaktır.

Entegreli Adım Motor Sürücü Devresi



Her şeyden önce bir adım motora ihtiyacımız var. İşimize en çok yarayacak olan adım motorunu eski 5 ¼ disket sürücülerinden kolayca sökebilirsiniz. Eğer bulamıyorsanız bozuk yazılardaki step motorları kullanabilirsiniz. Bulacağınız adım motoru 4,5, 6 ya da 8 kablolu olabilir. Uygulamamızda 5 kablolu adım motoru kullanacağız. Çalışma prensipleri aynı olduğundan kablo sayısının değişmesi yazacağımız programın mantığının değişmesi anlamına gelmeyecektir.



5 kablolu adım motorunun kablolarından biri vMotor dediğimiz ortak kablodur. Önemli olan bu kablonun hangisi olduğunu bulmaktır. Bunun için avometreninizi OHM ölçere getirin ve kabloların uçlarını ikişer ikişer ölçün. Tüm uçlar ile arasındaki direnç aynı olan kablo ortak vMotor kablosudur. Resimde gözüktüğü gibi diğer 4 kablo motor bobinlerine bağlıdır. Bu 4 kablonun da bir sırası vardır. Bu sırayıda deneme yanılma yöntemiyle bulmak mümkün olacaktır. Eğer bu kabloları yanlış sırada bağlarsanız motor dönmek yerine sadece titreyecektir. Yukarıda da bahsettiğim gibi motora adım attırmak için yapmamız gereken vMotor kablosuna +12 Volt verirken diğer bobinlere bağlı kablolara belli bir sıra ile toprak göndermektir.

Yukarıda bahsettiğimiz bu 4 kabloya toprak sinyalini göndermek için entegrenin 3, 4, 5 ve 6. bacaklarına sıra ile +5 Volt göndermemiz gerekiyor. Hâliyle bu + 5 Voltu paralel portun şeklinde görülen DATA pinlerinden çekeceğiz.

Herzamanki gibi DATA pinlerinden çıkış almak için OUT komutunu kullanıyoruz. DATA portundan 8 bitlik veri çıkışı alabiliyoruz. DATA portuna hiçbir veri göndermediğimiz zamanki değeri "00000000" dır. Dikkat ederseniz 8 tane "0" var. Örneğin, data portuna 25 değerini gönderelim. 25 değerinin ikilik sayı sisteminde karşılığı "00011001" dir. Bu durumda D4, D3 ve D0 pinlerine karşılık gelen lojik değerler "1" olduğundan o pinler +5 Volt olacaktır.
Örnek: OUT &h378,25

Buradaki &h378 ise paralel portunuzun taban adresidir. Biz şekilde D0, D1, D2, D3 ile gösterilen ilk dört data pinini kullanacağız. Burada OUT komutlarını peş peşe kullanamayız çünkü veriyi DATA pinlerine ufakta olsa belli zaman aralıkları ile göndermemiz gerekiyor. Bunun için her OUT komudunun arasına bekletmek için çok eski yöntemlerden biri olan FOR-NEXT döngüsünü yazıyoruz. Fakat dikkat etmeniz gereken nokta FOR-NEXT döngüsünü bitirdiğiniz sayıdır. Uygulamada "2000" kullanılmıştır. Aşağıdaki Visual Basic program örneği D0, D1, D2 ve D3 pinlerini sıra ile +5 Volt yapacak ve entegrenin bu pinlerine bağlı olan bacaklarının karşılıklarını toprağa çekecektir. Neticede motorunuz bir adım atmış olacaktır.

Uygulamada kullanılan adım motoru 1.8 dereceliktir. Motora attıracağınız her normal adımda 1.8 derecelik bir dönme elde edeceğiniz anlamına gelmektedir. Motorun 1 tur dönmesi için 200 adımda atması gerekmektedir. Motorun vMotor dışında kalan diğer 4 kablosuna göndereceğiniz sinyallere göre bu adımın yönünü ve açısını değiştirmeniz mümkün olacaktır. En basitinden motora ters adım attırmak için sinyalleri D3'ten D0'a doğru göndermeniz yeterli olacaktır.

Yarım Adım Metodu				
Değer	**D3**	**D2**	**D1**	**D0**
9	1	0	0	1
1	0	0	0	1
2	0	0	1	0
6	0	1	1	0
4	0	1	0	0
2	0	0	1	0
12	1	1	0	0
8	1	0	0	0

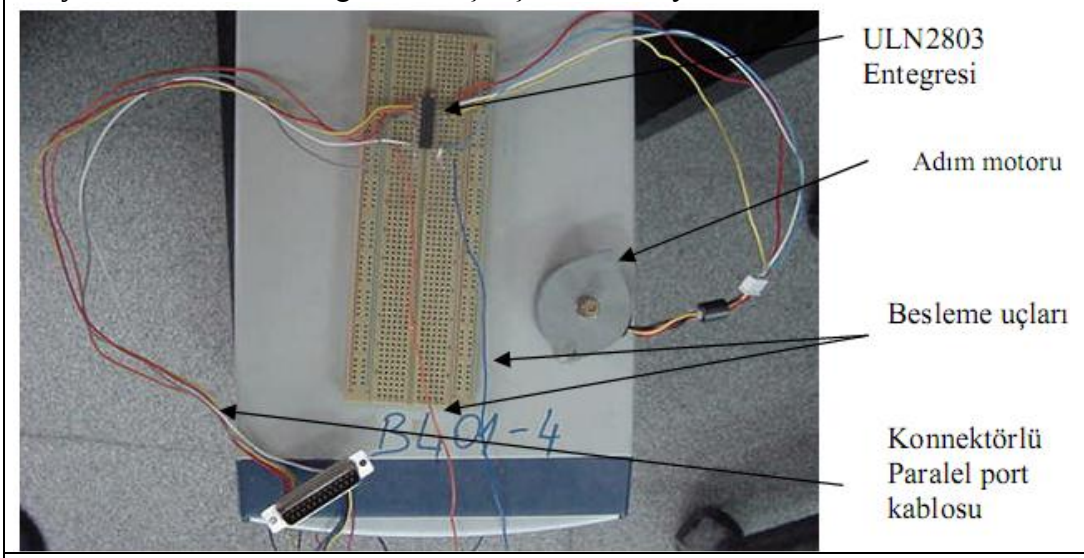
Tam Adım Metodu				
Değer	**D3**	**D2**	**D1**	**D0**
9	1	0	0	1
3	0	0	1	1
6	0	1	1	0
12	1	1	0	0

Dalga sürümü Adım Metodu				
Değer	**D3**	**D2**	**D1**	**D0**
1	0	0	0	1
2	0	0	1	0
4	0	1	0	0
8	1	0	0	0

İşlem Basamakları	**Öneriler**
➢ Kullanacağınız adım motorunun ➢ adımlama sırasını tespit ediniz.	➢ Öncelikle sargı direncini ölçerek **moto-run şase**sini bulmalısınız. Daha sonra kablolarına sırayla +5V vererek motorun aynı yönde dönmesini sağlayınız. Motorun dönmesini görmek için miline herhangi bir renkli bant parçası yapıştırabilirsiniz.
➢ Bilgisayarın paralel portundan ➢ breadboard üzerine bağlantı yap-mak için bir ucu paralel port konnektörüyle lehimlenmiş şekilde 50 – 100 cm arası bir kablo hazırlayınız.	➢

➢ ULN2803 sürücü entegresinin kullanım kılavuzunu temin ede-rek detaylıca inceleyiniz. Aşa-ğıda verilen kodları formunuza uyarlayınız.	➢ Bu entegre temin edilmesi çok kolay ve ➢ yaygın bir elemandır. Kullanım katalo-ğu ise İnternet ortamından bulunabilir.

➢ Breadboard üzerine devreyi kurunuz. Bunun için aşağıdaki fotoğraf size yardımcı olabilir. Programınızı çalıştırırak deneyiniz.



➢ Programınızın algoritmasını düşününüz ve çiziniz.

➢ Programınız için biri form tasarımı yapınız ve kullandığınız nesnelerin özelliklerini ayarlayınız.

➢ Program kodlarınızı yazınız. Yazdığınız programı deneyiniz.

Kodlar

```
Sub bekle(sayi as long)

Dim i as long

        For i=1 to sayi:next i

End sub

Command1_Click()

'Bu örnek program,  dalga sürümü sinyallerini, motora göndermektedir.
```

```
Dim adres as integer

        Adres=&h378

        OUT adres, 0 ‘Tüm data pinlerinin değerlerini “0” olacaktır.

                Bekle 2000

OUT adres,1 ‘D0 pininin lojik değeri “1” olacak ve +5 Volt yüklenecek.

        Bekle 2000 ‘Tanımladığımız BEKLE fonksiyonu ile bir sürelik gecikme
sağlıyoruz.

OUT adres,2 ‘D1 pininin lojik değeri “1” olacak ve +5 Volt yüklenecek.

        Bekle 2000 ‘Tanımladığımız BEKLE fonksiyonu ile bir sürelik gecikme
sağlıyoruz.

OUT adres,4 ‘D2 pininin lojik değeri “1” olacak ve +5 Volt yüklenecek.

        Bekle 2000 ‘Tanımladığımız BEKLE fonksiyonu ile bir sürelik gecikme
sağlıyoruz.

OUT adres,8 ‘D3 pininin lojik değeri “1” olacak ve +5 Volt yüklenecek.

        Bekle 2000 tanımladığımız BEKLE fonksiyonu ile bir sürelik gecikme
sağlıyoruz.

End Sub
```

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
1. Step motor sürücü kartını hazırlayabildiniz mi?		
2. Formu tasarlayıp kodları yazabildiniz mi?		
3. Programı denemek amacıyla çalıştırabildiniz mi?		

DEĞERLENDİRME

Değerlendirme sonunda "Hayır" şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız "Evet" ise "Ölçme ve Değerlendirme" ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları dikkatlice okuyunuz ve doğru seçeneği işaretleyiniz.

1. Visual Basic programlama dilinde paralel port uygulamalarını yapabilmek için hangi dosyaya ihtiyaç vardır?
A) parallel.ocx
B) inpout32.dll
C) port32.dll
D) inpout.dll

2. Aşağıdaki komutlardan hangisi doğrudur?
A) Out val(378) , veri
B) Outport(H256), 255
C) Out Val (&H379), 255
D) Out (H378), veri

3. Visual Basic'te paralel portu kullanmak için gerekli dll dosyası hangi bölümde koda eklenir?
A) Form
B) Özellikler
C) Modül
D) General

4. Paralel portun çıkış iğnelerinden alınan gerilim kaç volt'tur?
A) 5V
B) 2V
C) 12V
D) V

5. Aşağıdaki programın çıktısı nedir?

```
Sub Command1_click()
Veri=&HAA
Out Val (H378) , veri
End Sub
```

A) 10101011
B) 01010101
C) 10100101
D) 10101010

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ–3

AMAÇ

Paralel port donanımı yoluyla sayısal giriş değerlerini kontrol edebileceksiniz.

ARAŞTIRMA

- Farklı programlama dillerinde paralel port kullanarak dış ortamdaki verilerin alımı nasıl yapılmaktadır?

3. PARALEL PORTTAN GİRİŞ ALMAK

Paralel portu kullanarak dışarıya veri gönderebildiğimiz gibi dış ortamdaki herhangi bir veriyi de paralel port üzerinden alabiliriz. Durum ve denetim yazmaçları üzerinden bitler hâlinde veri okumak mümkündür. Harici ortamdan almak istediği-miz veriler bit şeklinde olabileceği gibi 1 bayt'lık düzende de olabilir. **Eğer 1 bayt'lık bir veri okunmasına ihtiyaç duyuluyorsa bunu üç yolla:**

- Durum portundan ardı ardına 4 bit okuyup bunları birleştirerek,
- Durum portundan 4 bit, denetim portundan 4 bit okuyarak birleştirmekle,
- Çift yönlü, EPP ve ECP port modlarını kullanarak veri hattından okuyarak **alınabilir.**

3.1. Paralel Port Durum Yazmacından Giriş Almak

Durum yazmacının esas işlevi, yazıcının bilgisayara kendi durumu ile ilgili bilgileri aktarmasına aracılık etmektir. Yazıcının "Kâğıt yok." bilgisini göndermesi ya da o anda başka bir yazdırma görevi yapıyor ise "Meşgul" bilgisini bilgisayara bildirmesi bunlara örnek olarak verilebilir. Aşağıdaki tablo durum yazmacıyla ilgili bir fikir vermektedir.

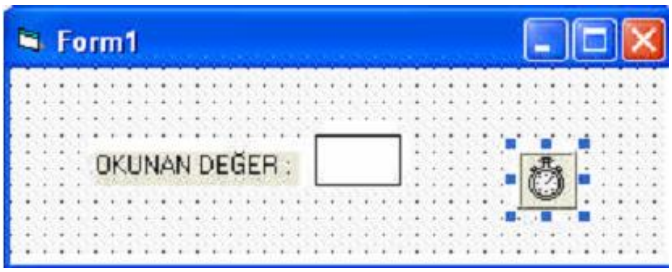
Temsil	İsim (görev)	Pin	Okunan Değer	İkilik karşılığı
S7	Busy (Meşgul)	11	255	11111111
S6	Ack (Hazır)	10	63	00111111
S5	NoPaper(Kâğıt yok.)	12	95	01011111
S4	Selected (Seçili)	13	111	01101111
S3	Error (Hata)	15	119	01110111
S2-S1-S0	Tanımsız	-	-	-

Tablo 3.1: Durum yazmacı

Hiçbir giriş yapılmadığı taktirde okunan değer 127 (01111111)'dir çünkü deney kartı üzerinde tüm girişler (girişler) 4,7K değerinde dirençlerle 5V gerilime bağlanmıştır. **"Meşgul"** biti (11no'lu uç) terslendiği için bu bit 0 olarak okunur. Paralel port bağlacında temsil edilmeyen ilk 3 bit donanım olarak 1 görülür. Dolayısıyla girişe bağlanan 5 butona sırayla basıldığında yukarıdaki tabloda verilen değerler okunacaktır.

Not: Bazı bilgisayarlarda okunan değerler farklılık gösterebilir. Örneğin, 119 yerine 118 okunabilir. Bunu nedeni tanımsız olan ilk 3 bitin durumudur. EPP modunda ilk bit zaman aşımı biti olarak atanır ve 0 değerini alır. Buna göre değerler farklı çıkar.

Örnek: Deney kartımıza bulunan ve durum portu uçlarına bağlı olan butonlara bastığımızda ekranda bu giriş değeri gösteren bir program yazalım. Port üzerinden okuma işlemi yaptırılırken sürekli olarak yazmaç iğnelerinin dinlenmesi gerekir. Bunun için bir döngü kurabilirsiniz ancak en sağlıklı yöntem "Timer" nesnesinin kullanılmasıdır.



Resim 3.1: Örnek form

Formun üzerine timer nesnesi yerleştirdikten sonra interval özelliğini 100 olarak belirleyelim. Bunun anlamı her 100 mili/saniye aralıklarla timer nesnesi çalışacaktır.

Tüm örneklerde olduğu gibi modülü aynı şekilde projenize ekledikten sonra kod yazımına geçelim. Kodlarımızı timer nesnesine alt yordam olarak ekleyelim. Böylelikle her 100 mili/saniyede bir kodlarımız çalışacak ve durum yazmacındaki veriler alınacaktır.

Paralel porttan bilgi okumak için "inpout32.dll" kütüphanesinin "INP ()" komutu kullanılır. Okunan değer herhangi bir değişkene aktarılarak program içinde kullanılabilir. Bu komutun kullanım formatı aşağıdaki gibidir.

Değişken adı=INP (&adres)

Bu örneğimizde durum portundan okuma işlemi yaptığımızdan dolayı durum yazmacının adresini INP komutu içerisine yazarız.

```
Private Sub Timer1_Timer()

oku = Inp(&H379) '379H durum yazmacı adresinden okuma işlemi yap ve oku değişkenine aktar.

Text1.Text = oku 'okunan değeri metin kutusuna yazdır.

End Sub
```

Yukarıdaki kod satırlarını yazdıktan sonra programımızı çalıştıralım ve deney kartı üzerindeki butonlara sırayla basarak aşağıdaki tabloyu doldurunuz.

Temsil	Pin	Okunan Değer	İkilik karşılığı
S7	11		
S6	10		
S5	12		
S4	13		
S3	15		

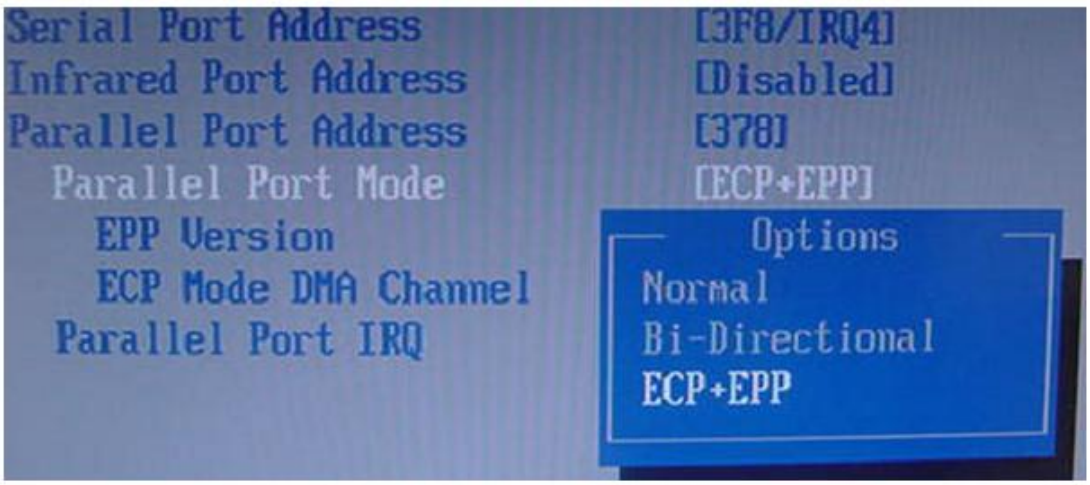
3.2. Paralel Port Veri Yazmacından Giriş Almak

Veri yazmacı normalde yazıcıya veri gönderilmesi için çıkış amaçlı olarak kullanılır. Bazı durumlarda dış ortamdan 8 bitlik veri alınması gerekebilir. Eski tip paralel portlar bu gibi durumlar için veri portundan veri girişine imkân tanımaz iken sonradan geliştirilen ve günümüz ana kartlarında kullanılan paralel portlar ile bu işlem gerçekleştirilebilir.

Günümüz ana kartlarındaki paralel portlarda 1.öğrenme faaliyetinde "1.2.Paralel Port Modları" başlığı altında bahsedilen "çift yönlü (bi-directional)" adı verilen çalışma modu bulunmaktadır. Bilgisayarımızın paralel portu genellikle fabrika ayarları olarak **"normal"** mod olarak ayarlanmıştır. Veri yazmacından giriş almak için öncelikle yapılması gereken işlem bilgisayarın BIOS programından paralel port modunun çift yönlü (bi-directional)

duruma getirilmesi gerekmekte-dir. Bazı BIOS programlarında çift yön (bi-directional) modu ayrı bir seçenek hâlinde sunulmamıştır. Bu durumda EPP ya da ECP modlarından birisi seçilmelidir.

Aşağıdaki şekilde örnek bir BIOS ekranında paralel port ayarları görülmektedir:



Resim 3.2: Bios ekranı

Çift yönlü portlarda denetim yazmacının 5.biti (C4 – enable) yön biti olarak kullanılmaktadır. Hatırlayacağınız gibi denetim yazmacının ilk 4 biti konnektör pinlerine bağlı olarak kullanılmaktadır. C4 bitinin konnektör üzerinde fiziki bir karşılığı bulunmamaktadır. Dolayısıyla bu bitin kullanımı program üzerinden olmaktadır. C4 biti mantık 1 yapılırsa veri yazmacı giriş, mantık 0 yapılırsa çıkış olarak ayarlanır.

Örnek:

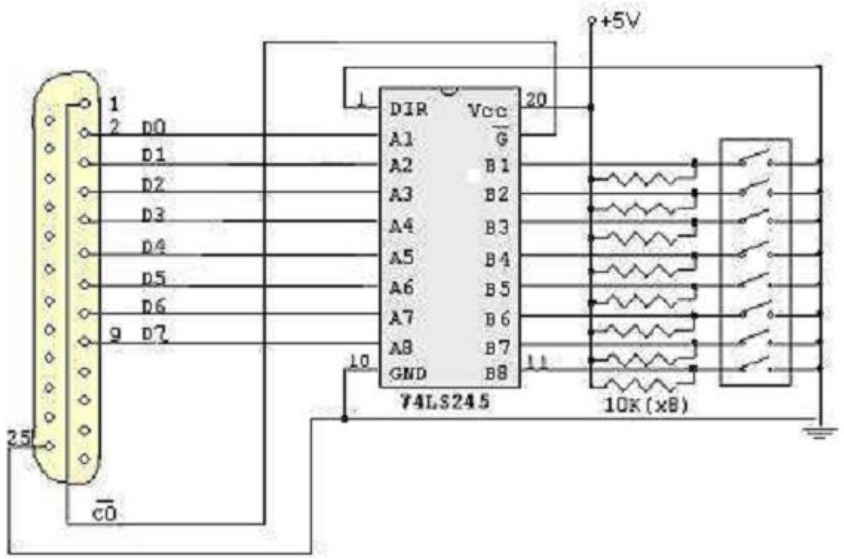
Bu örneğimizde paralel portumuzun veri yazmacına bağladığımız bir buton yardımı ile giriş alalım ve değeri formumuzda görelim.

Bu örneği gerçekleştirmek için öncelikle aşağıdaki devreyi breadboard üzerinde kuralım. Devrede kullanılan 74LS245 entegresi üç durumlu iki yönlü sekizli (octal) tampon olarak görev yapmaktadır. 74LS245 entegresi aşağıdaki doğruluk tablosuna göre işlem yapmaktadır.

GİRİŞ		ÇIKIŞ
G	DIR	
L	L	B'den A'ya aktarım
L	H	A'dan B'ye aktarım
H	X	Yalıtımda

Tablo 3.2: 74LS245 doğruluk tablosu

Doğruluk tablosundan da anlaşılacağı üzere 74LS245 entegresinin B'den A'ya aktarım yapabilmesi için G ve DIR iğnelerinin her ikisinin de L (low-düşük) yapılması gerekmektedir. DIR (direction) ucu toprak hattına verilerek L (low-düşük) yapılır. G ise paralel port konnektörün de 1no'lu iğneye bağlanmıştır. 1 no'lu iğne denetim yazmacında C0 bitinin karşılığıdır ve terslenmiştir.



Resim 3.3: Buton devresi

Bu işlemden sonra bilgisayarımızın BIOS' una girip paralel portumuzu "çift yönlü (bidirectional)" moda ayarlayalım. Daha sonra aşağıdaki gibi basit bir form ta-sarlayarak program yazımına geçebiliriz.

Resim 3.4: Örnek form

Programımızda 74LS245 entegresinin G ucunu L (low) yapmak için C0 bitini veri yazmacını giriş olarak ayarlamak için ise C4 bitini mantık 1 yapmalıyız.

C0	C1	C2	C3	C4	C5	C6	C7	
1	0	0	0	1	0	0	0	=17

Resim 3.5: Durum yazmacının bitlerinin aldıkları değerler

Bunun için elde ettiğimiz 17 bilgisini form Load (yükle) kısmında denetim yazmacına gönderelim. Form Load kısmına yazarak programımız çalıştırıldığında bu ayarın yapılmasını sağlamaktır.

```
Private Sub Form_Load()

Out Val("&H37A"), Val(17) 'denetim yazmacının 5.biti mantık 1 yapılır.

End Sub

Private Sub Command1_Click()

oku = Inp(&H378) 'veri yazmacı okunur.

Text1.Text = oku 'okunan değer metin kutusuna yazdırılır.

End Sub
```

Butonumuzun kod sayfasına ise veri yazmacından değer okuyarak metin kutusuna yazılmasını sağlayan kod satırlarımızı ekleyerek kod yazımını sonlandıra-lım. Son olarak programı çalıştıralım ve breadboard üzerinden butonlara basarak metin kutusunda okunan değerin doğruluğunu kontrol edelim.

UYGULAMA FAALİYETİ

Aşağıdaki işlem basamaklarına göre uygulama faaliyetini yapınız.

Aşağıdaki form üzerinde bir kimyasal sıvının endüstriyel ortamda işlenmesi görülmektedir. Bu işlem şu şekilde çalışmaktadır:

- Dolum vanası butonuna basılarak işlem başlatılır ve tanka kimyasal sıvı girişi yapılır.
- Tank dolduğunda vana otomatik olarak kapanır ve ısıtıcı çalışır.
- Sıcaklık sensöründen kaynama bilgisi geldiğinde ısıtıcı kapanır.
- Boşaltım vanası otomatik olarak açılarak tank içindeki kimyasal sıvı boşaltılır.

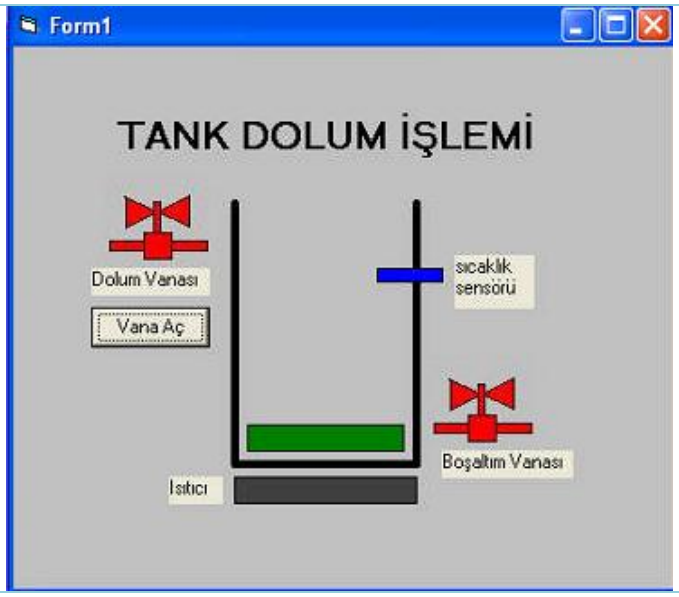
Bu uygulama için deney kartından yaralanılacaktır. Buna göre;

Sıcaklık sensörü: Deney kartındaki durum portu butonlarından biri seçilecektir. Sıcaklık sensörünün göndereceği kaynama bilgisi temsili olarak butona basılarak verilecektir.

Dolum vanası: Deney kartındaki 1. veri yazmacı ledi seçilecektir. Vana form üzerinden açıldığında yeşile dönecek ve led1 yakılarak temsili olarak vananın açıldığını gösterilecektir. Tank dolduğunda led1 sönecek ve vana tekrar kırmızıya dönecektir.

Boşaltım vanası: Deney kartındaki 2. veri yazmacı ledi seçilecektir. Vana form üzerinden açıldığında yeşile dönecek ve led2 yakılarak temsili olarak vananın açıldığını gösterilecektir. Tank dolduğunda led2 sönecek ve vana tekrar kırmızıya dönecektir.

Isıtıcı: Deney kartındaki 3. veri yazmacı ledi seçilecektir. Isıtıcı sıcaklık sensöründen kaynama bilgisi alınarak (butona basıldığında) çalıştığında kırmızıya dönecek ve led3 yakılarak temsili olarak ısıtıcının çalıştığı gösterilecektir. Isıtıcı durduğunda led3 sönecek ve ısıtıcı tekrar siyaha dönecektir.

Ekran Görüntüsü	
	
İşlem Basamakları	
1. Programınız için form tasarımı yapınız.	
2. Form üzerindeki nesne özellikleri ayarlayınız.	
3. Programın algoritmasını geliştiriniz.	
4. Program kodlarınızı yazınız.	
5. Programınızı çalıştırarak deneyiniz.	

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
1. Formu tasarlayabildiniz mi?		
2. Kodları yazabildiniz mi?		
3. Programı denemek amacıyla çalıştırabildiniz mi?		

DEĞERLENDİRME

Değerlendirme sonunda "Hayır" şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız "Evet" ise "Ölçme ve Değerlendirme" ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları cevaplayarak bu faaliyette kazandığınız bilgileri ölçünüz.

1. Veri girişi yapmak için hangi komut kullanılır?
   A) Inpout()
   B) Inp()
   C) DataInput ()
   D) In()

2. Aşağıdaki ifadelerden hangisi yanlıştır?
   A) Visual Basic'te paralel port kullanmak için inpout32.dll dosyası gereklidir.
   B) Veri girişi yaparken tampon (buffer) entegresi kullanmak portu korur.
   C) Normal modda veri yazmacından giriş yapılamaz.
   D) Visual Basic'te paralel port kontrolünde denetim yazmacından giriş yapılamaz.

3. Durum yazmacının hangi biti terslenmiştir?
   A) 0
   B) 2
   C) 5
   D) 7

4. Aşağıdaki seçeneklerde verilen satırlardan hangisi doğrudur?
   A) Z =Inp(oku, &378)
   B) Oku ::Inp (&H38A)
   C) Oku= Inp (&H378)
   D) Oku= In(378), 8

5. Aşağıdakilerden hangisi paralel port modlarından birisi değildir?
   A) Standart
   B) EPP
   C) EPD
   D) ECP

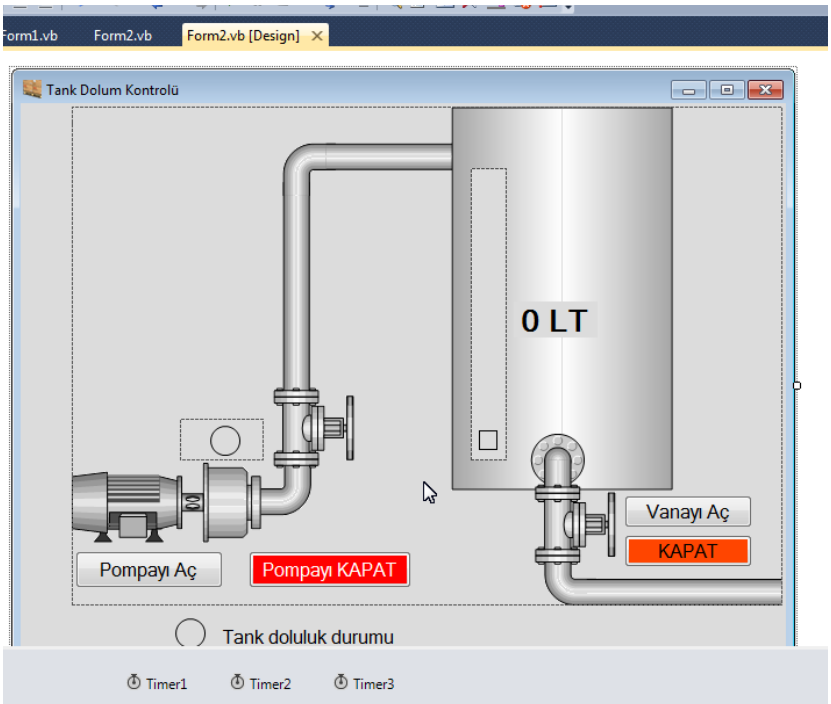
DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise Modül Değerlendirmeye geçiniz.

MODÜL DEĞERLENDİRME

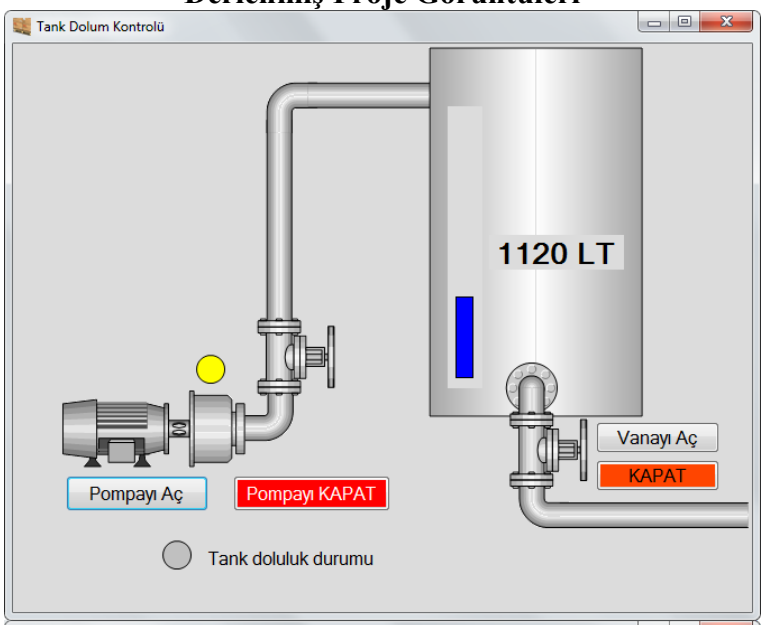
Soru	Tank dolum tesisinin kontrolüne ait program yazılacaktır. Tan dolum kontrolü yapılırken oluşacak hatalar Flaşör şekille ve yazılarla gösterilecek. Tankın seviye durumları ve pompa motorunun çalıştığını gösteren şekillere yer verilecektir.

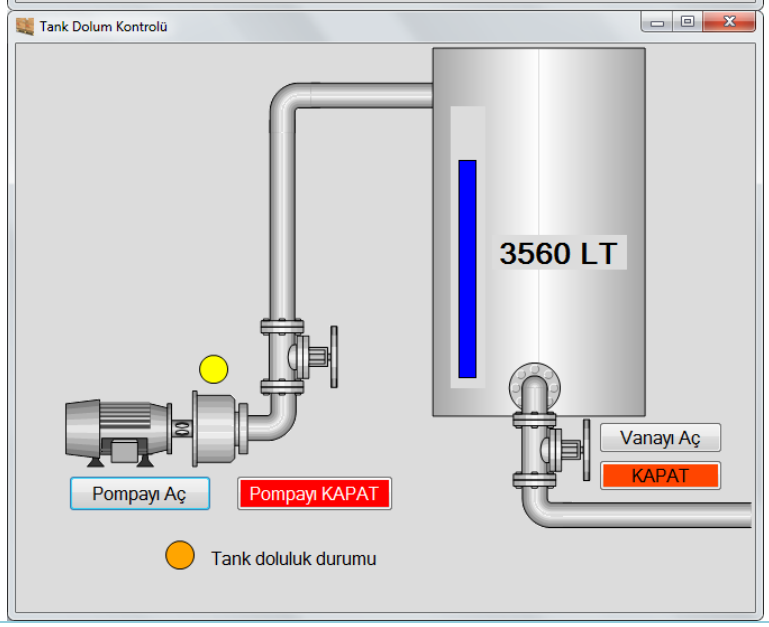
Ekran Görüntüleri

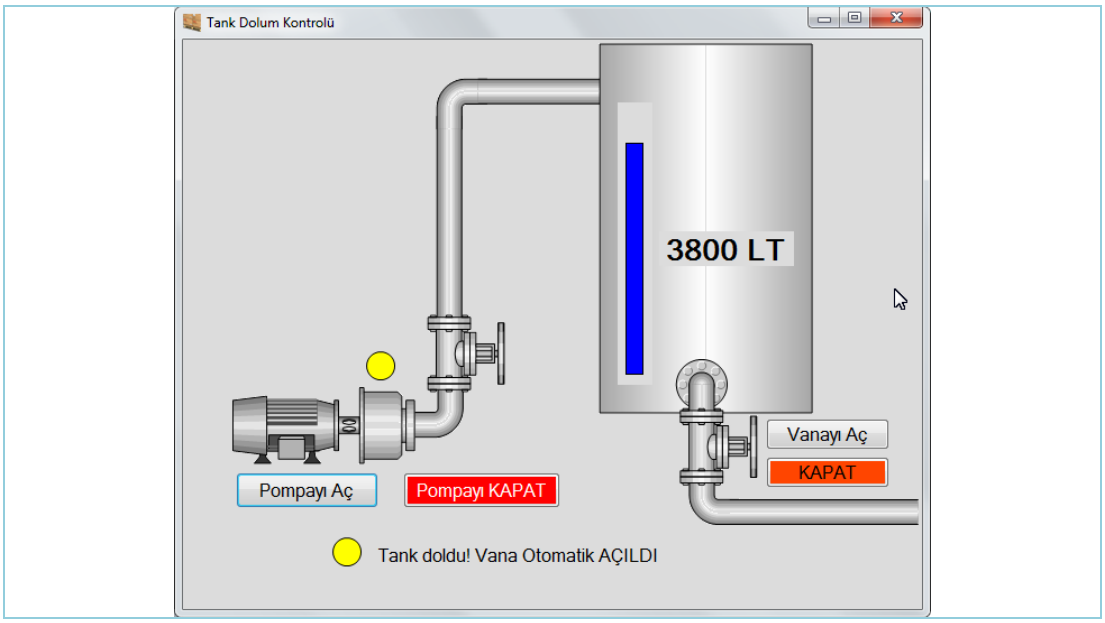


Micrsoft Visual Basic Proje Ekranı

Derlenmiş Proje Görüntüleri







Aşağıda verilen değerlendirme ölçütleriyle kendinizi değerlendiriniz.

Değerlendirme Ölçütleri		Evet	Hayır
1.	Formu tasarlayabildiniz mi?		
2.	Timer ayarlarını yapabildiniz mi?		
3.	Gerekli kodları yazabildiniz mi?		
4.	Programlarınız doğru ve hatasız çalıştı mı?		

DEĞERLENDİRME

Değerlendirme sonunda "Hayır" şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız "Evet" ise bir sonraki modüle geçmek için öğretmeninize başvurunuz.

CEVAP ANAHTARLARI

ÖĞRENME FAALİYETİ-1'İN CEVAP ANAHTARI

1	**C**
2	**C**
3	**D**
4	**B**
5	**A**
6	**C**

ÖĞRENME FAALİYETİ-2'NİN CEVAP ANAHTARI

1	**B**
2	**C**
3	**C**
4	**A**
5	**D**

ÖĞRENME FAALİYETİ-3'ÜN CEVAP ANAHTARI

1	**B**
2	**D**
3	**A**
4	**C**
5	**C**

KAYNAKÇA

- Microsot MSDN Library
- AXELSON Jan, **Her Yönüyle Paralel Port**, Bileşim Yayıncılık & Era Bilgi Sistemleri ve Yayıncılık, Mart – 2000.
- ÖZDEVECİ Murat, **Bilgisayar ile Kumanda Teknikleri**, M.E.B. Hizmet İçi Eğitim Seminer Notları, İzmir - 2005.